Formulations PLNE pour l'ordonnancement des chaînes sur une machine

Philippe Baptiste et Ruslan Sadykov

École Polytechnique, CNRS LIX, F-91128 Palaiseau
Philippe.Baptiste@polytechnique.fr, sadykov@lix.polytechnique.fr

Mots-clés : ordonnancement juste-à-temps, programmation linéaire en nombres entiers.

1 Introduction

Le problème considéré traite de l'ordonnancement de tâches par des radars aéroportés. Le modèle, sur lequel nous travaillons, a été proposé par Winter et Baptiste [3] : un ensemble $N = \{1, \ldots, n\}$ de tâches doit être exécuté sur une machine. Chaque tâche consiste en une chaîne d'opérations identiques, $(O_{i0}, O_{i1}, \ldots, O_{i,n(i)})$, qui doivent être exécutées dans cet ordre sur un horizon de temps $[0, h]$. On définit $N(i) = \{1, \ldots, n(i)\}$. On suppose que l'exécution d'une opération ne peut être interrompue, et que la machine ne peut exécuter plus d'une opération à la fois. La durée d'exécution de chaque opération d'une tâche i est égal à $p_i > 0$. Soit S_{ij} la date de début de l'opération O_{ij}. On suppose que, pour chaque tâche $i \in N$, S_{i0} est fournie en entrée du problème.

Pour ce problème, deux opérations consécutives d'une même tâche $O_{i,j-1}$ et $O_{i,j}$, doivent être idéalement ordonnancées de telle façon à ce que $S_{ij} - S_{i,j-1} = l_i$, l_i étant une constante fournie en entrée du problème. Cependant, il peut ne pas y avoir d'ordonnancement idéal à cause des contraintes de non-chevauchement des tâches. Winter et Baptiste [3] ont introduit une fonction de coût pour pénaliser un écart par rapport à la distance optimal l_i entre deux opérations consécutives d'une même tâche : pour chaque tâche $i \in N$, on a $\delta_i(x) = \max\{\alpha_i(l_i - x), \beta_i(x - l_i)\}$. Ainsi, le problème consiste à chercher un ordonnancement réalisable $\{S_{ij}\}_{i \in N, j \in N(i)}$ qui minimise la somme des pénalités, $F = \sum_{i \in N} \sum_{j \in N(i)} \delta_i(S_{ij} - S_{i,j-1})$. On suppose de même que toutes les données sont entières sauf $\{\alpha_i, \beta_i\}_{i \in N}$.

Ce problème est une généralisation du problème juste-à-temps sur une machine $1||\sum E_j + T_j$. Ce dernier étant NP-complet au sens fort [1], le problème étudié ici l'est aussi.

2 Formulation naïve

Nous proposons tout d'abord une formulation naïve (TI) du problème par un Programme Linéaire en Nombres Entiers (PLNE). Nous reprenons l'idée de formulation indexée par le temps pour les problèmes d'ordonnancement sur une machine [2]. On définit $H = \{0, 1, \ldots, h - 1\}$. La variable binaire X_{ijt}, $i \in N$, $j \in N(i)$, $t \in H$, prend la valeur 1 si et seulement si l'opération O_{ij} est commencée à t. La variable continue S_{ij}, $i \in N$, $j \in N(i)$, est égale à la date de début de l'opération O_{ij}. La variable continue W_{ij}, $i \in N$, $j \in N(i)$, représente la valeur $\delta_i(S_{ij} - S_{i,j-1})$.

$$
(\mathrm{TI}) \quad
\begin{aligned}
&\min \sum_{i \in N} \sum_{j \in N(i)} W_{ij} && (1)\\
&s.t. \sum_{t=0}^{h-p_i} X_{ijt} = 1, \quad i \in N, j \in N(i) \cup \{0\}, && (2)\\
&\quad \sum_{i \in N} \sum_{j \in N(i) \cup \{0\}} \sum_{t'=\max\{t-p_i+1, 0\}}^{t} X_{ijt'} \leq 1, \quad t \in H, && (3)\\
&\quad S_{ij} = \sum_{t \in H} t \cdot X_{ijt}, \quad i \in N, j \in N(i) \cup \{0\}, && (4)\\
&\quad S_{i,j-1} + p_i \leq S_{ij}, \quad i \in N, j \in N(i), && (5)\\
&\quad W_{ij} \geq \alpha_i(l_i - S_{ij} + S_{i,j-1}), \quad i \in N, j \in N(i) && (6)\\
&\quad W_{ij} \geq \beta_i(S_{ij} - S_{i,j-1} - l_i), \quad i \in N, j \in N(i), && (7)\\
&\quad X_{ijt'} \in \{0, 1\}, \quad i \in N, j \in N(i) \cup \{0\}, t \in H. && (8)
\end{aligned}
$$

3 Nouvelle formulation

Nous décrivons une autre formulation du problème par un PLNE. Soit $n_i(t)$, $i \in N$, $t \in \Delta_i$, le nombre d'opérations de la tâche i commencées dans l'intervalle $[t - l_i + 1, \ldots, t]$ (voir la Figure 1). Soit Δ_i, $i \in N$, un sous-intervalle de $[0, h]$ tel que $t \in \Delta_i$ si $n_i(t) > 0$ ou $S_{i0} \le t \le S_{i,n(i)}$. Il est facile de voir que $\Delta_i = \{S_{i0}, S_{i0}+1, \ldots, S_{i,n(i)} + l_i - 1\}$, $i \in N$. On introduit maintenant une nouvelle fonction de pénalité : $\gamma_i(t) = \max\left\{\alpha_i\big(n_i(t) - 1\big), \beta_i(1 - n_i(t))\right\}$.

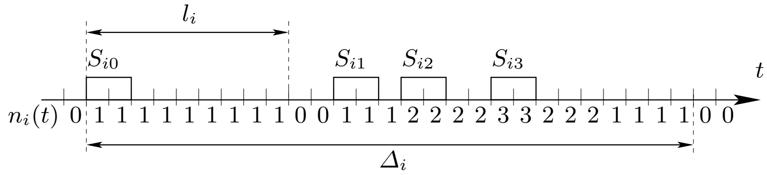

Fig. 1. Ordonnancement de la tâche i ($n(i) = 3$, $p_i = 2$, $l_i = 9$)

Théorème 1. *Deux fonctions de pénalité δ et γ induisent la même valeur pour un ordonnancement donné :* $\sum_{i \in N} \sum_{j \in N(i)} \delta_i(S_{ij} - S_{i,j-1}) = \sum_{i \in N} \sum_{t \in \Delta_i} \gamma_i(t)$.

On va maintenant ajuster la formulation (TI) en utilisant l'équivalence entre les fonctions de pénalité. On envisage un horizon de temps prolongé pour chaque tâche $i \in N$: $H_i = \{S_{i0}, \cdots, h - p_i + l_i\}$. La variable entière X_{it}, $i \in N$, $t \in H$, est maintenant égale au nombre des opérations de la tâche i commencées avant ou à t. La variable continue W_{it}, $i \in N$, $t \in H_i$, représente la valeur $\gamma_i(t)$. Le problème principal lié à la fonction de pénalité alternative est que les dates de début $S_{i,n(i)}$, $i \in N$, ne sont pas connus a priori. Donc, on a besoin d'utiliser des variables additionnelles E_{it}, $i \in N$, $t \in H_i$, qui indiquent si $t \in \Delta_i$. La formulation PLNE alternative est

$$
\text{(TIA)} \quad
\begin{aligned}
&\min \sum_{i \in N} \sum_{t \in H_i} W_{it} && (9)\\
&s.t.\ X_{i,S_{i0}} = 1, \quad X_{i,S_{i0}-1} = 0, \quad X_{i,h-p_i} = n(i) + 1, \quad i \in N. && (10)\\
&\quad X_{i,t-1} \le X_{it}, \quad i \in N,\ t \in H \setminus \{0\}, && (11)\\
&\quad \sum_{i \in N} X_{it} - \sum_{i \in N:\ t - p_i \ge 0} X_{i,t-p_i} \le 1, \quad t \in H, && (12)\\
&\quad E_{it} \ge X_{i,t-l_i+p_i} - X_{i,t-l_i},\ i \in N,\ t \in H_i,\ t \ge S_{i0} + l_i, && (13)\\
&\quad E_{i,t} \le E_{i,t-1} \le 1, \quad i \in N,\ t \in H_i \setminus \{S_{i0}\}, && (14)\\
&\quad W_{it} \ge \alpha_i(X_{it} - X_{i,t-l_i} - E_{it}), \quad i \in N,\ t \in H_i, && (15)\\
&\quad W_{it} \ge \beta_i(E_{it} - X_{it} + X_{i,t-l_i}), \quad i \in N,\ t \in H_i, && (16)\\
&\quad X_{it} \in \mathbb{Z}_+, \quad i \in N,\ t \in H. && (17)
\end{aligned}
$$

Soit ν_F la valeur d'une solution optimale de la relaxation PL d'une formulation (F).

Théorème 2. *Si les dates de début $\{S_{i,n(i)}\}_{i \in N}$ des dernières opérations de chaque tâche sont fixées, alors $\nu_{TI} \le \nu_{TIA}$.*

Le Théorème 2 est faux dans le cas général. Néanmoins, ce théorème suggère que la formulation (TIA) est plus efficace que la formulation (TI). Les expérimentations numériques sur les instances générées aléatoirement ont confirmé cette hypothèse. En outre, on a, en moyenne, $\nu_{TI}/\nu_{TIA} \approx 0.26$. En utilisant la formulation (TIA) nous avons pu résoudre optimalement de nombreuses instances réelles de taille moyenne (9-18 tâches, 50-150 opérations, la longueur de l'horizon est 450-900) générées en collaboration avec le département d'ingénierie de Thales.

Références

1. M.R. Garey, R.E. Tarjan, G.T. Wilfong (1988), One-processor scheduling with symmetric earliness and tardiness penalties, *Mathematics of Operations Research* 13 :330–348.
2. J.P. Sousa, L.A. Wolsey (1992), A time-indexed formulation of non-preemptive single-macine scheduling problems, *Mathematical Programming* 54 :353-367.
3. E. Winter, Ph. Baptiste (2007), On Scheduling a Multifunction Radar, accepted for publication in *Aerospace Science and Technologies.*